Design and Implementation of Time Efficient Trajectories for an Underwater Vehicle

M. Chyba, T. Haberkorn [1]

University of Hawaii at Manoa, Mathematics Department, 2565 McCarthy Mall, Honolulu, HI 96822

R.N. Smith [1]

University of Hawaii at Manoa, Ocean and Resources Engineering, Honolulu, HI 96822

S.K. Choi [2]

University of Hawaii at Manoa, Autonomous System Laboratory, College of Engineering, Honolulu, HI 96822

Abstract

This paper discusses control strategies adapted for practical implementation and efficient motion of underwater vehicles. These trajectories are piecewise constant thrust arcs with few actuator switchings. We provide the numerical algorithm which computes the time efficient trajectories parameterized by the switching times. We discuss both the theoretical analysis and experimental implementation results.

Key words: Autonomous Underwater Vehicles, Optimal Control, Numerical Algorithm, Trajectory Planning.

---

1 Introduction

Underwater vehicles have been designed to perform a multitude of tasks and play many roles. From side-scan sonar to water sampling, their use in ocean

[1] Research supported in part by NSF grants DMS-030641, DMS-0608583
[2] Research partially supported by ONR Grant N00014-03-1-0969, N00014-04-1-0751, N00014-04-1-0751

research has gone from occasional to necessity. Whether they are tethered, towed or autonomous; torpedoes, gliders or robot fish, we must develop control strategies that govern their motions. Traditionally, autonomous underwater vehicles (AUV's) have taken the role of performing the long data gathering missions in the open ocean with little to no interaction with their surroundings [22]. The AUV is used to find the shipwreck, and the remotely operated vehicle (ROV) handles the up close exploration. AUV mission profiles of this sort are best suited through the use of a torpedo shaped AUV [1], such as WHOI's REMUS and MIT's Odyssey, since straight lines and minimal ($0°$ - $30°$) angular displacements are all that are necessary to perform the transects and grid lines for these applications. However, the torpedo shape AUV lacks the ability to perform low-speed maneuvers in cluttered environments, such as autonomous exploration close to the seabed and around obstacles [22,24]. Also a torpedo shape is easy to control along straight lines but would perform poorly if asked to follow a fast moving, agile target. This inability of the torpedo shape AUV is currently remedied through the use of an ROV. This approach has its advantages and disadvantages. Note though, more and more autonomous vehicles are being designed to take over these jobs from the ROV's [24]. However, we should not let application drive the limits of the theory or vise versa. Water provides a medium in which all six degrees of freedom are obtainable and readily accessible without the necessity to constantly provide lift. This gives us a configuration and operation space for our vehicle where we can exploit all six degrees of freedom. This has motivated the research and development of AUV's and their control systems which have the ability to take over more responsibilities in ocean research. As we give a vehicle more responsibilities, assuming all else is constant, it will require an increase in efficiency. From a practical point of view, efficiency is measured time or energy consumption. Here we address the time minimum problem. This is a first step toward minimizing a combination of both time and energy consumption along a given trajectory. The work presented here forms the foundation of an algorithm to compute efficient trajectories based on the vehicle's demands. Emergency avoidance places a heavier weight on time minimization, while long duration observation missions will require more energy efficiency. This work is currently under investigation. From a mathematical point of view, underwater vehicles belong to the class of simple mechanical systems; their Lagrangian is of the form kinetic energy minus potential energy. They can be characterized by differential geometric properties, see [5] for a recent treatment of geometric control for mechanical systems. Based on these geometric features [17,18] examines the stabilizing effect of gravity on the motion of a submerged rigid body in an unbounded ideal fluid . Without a doubt, the geometric framework is the correct architecture to study this application and exploit the inherent nonlinearity of the system. From a theoretical point of view, the time minimum trajectory planning problem for a submerged rigid body is addressed in [8–12]. These papers mainly focus on the conditions for an extremal to be singular. Even in the ideal fluid case, these extremals can be very complex and even contain an infinite number of actuator switches to acheive optimality. Trajectories such as these are impossible to realize with an AUV. Thus, we set out to create trajectories which can be implemented onto a test-bed AUV and are also time efficient. Moreover, we consider the real fluid case. From preliminary studies, it is clear that the complexity of the equations, due mainly to external forces, is such that we must consider numerical solutions. We assume the errors in the numerical computations are negligible with respect to errors in the approximation of the hydrodynamic model. Later, we present the algorithm used to compute the efficient implementable trajectories. In [3,4] the author's approach provides continuously varying controls as minimization solutions. This is a major inconvenience for practical implementation. On our test-bed vehicle, orientation, depth and open loop control are updated every 30ms. A continuously evolving control for any substantial time would exceed the onboard data storage limits. For this reason we consider a strategy based on piecewise constant controls. This strategy is designed to have a small number of changes or switches. For experimental testing, our efficient trajectories are implemented onto a spherical vehicle which is near neutrally buoyant and with center of gravity (C_G) very close to the center of buoyancy (C_B) (with respect to vehicle diameter). There is almost no preference of direction or orientation for movement, giving us a very controllable and versatile vehicle. The vehicle's design allows for simple drag estimations to give accurate coefficients, as well as exploits symmetry in the control theory through its geometry. However, the spherical vehicle gives virtually no resistance or restoration in yaw, and thus requires a good understanding of the vehicle's dynamics and thrusters in order to control it well. Due to this sensitivity, we run a feedback controller on the yaw component during testing. Our experimentation began with pure motion and concatenated pure motion trajectories and we obtained excellent results [7]. From these experiments we were able to fine tune our theoretical model and move on to implement our computed time efficient trajectories. Now we have successfully demonstrated the implementability and efficiency of the designed trajectories in many experiments. This capability of implementation allows us to stretch geometric control theory to its maximum potential for underwater applications and many other nonlinear mechanical control systems.

2 Equations of Motion

In [6] the equations of motion for a controlled rigid body immersed in a real fluid are introduced. Here briefly state the assumptions and equations. The position and orientation of a rigid body are identified with an element of $SE(3)$: (b, R), where $b = (b_1, b_2, b_3)^t \in I\!R^3$ denotes the position vector of the body and $R \in SO(3)$ is a rotation matrix describing the orientation of the body. The translational and angular velocities in the body-fixed frame are denoted by $\nu = (\nu_1, \nu_2, \nu_3)^t$ and $\Omega = (\Omega_1, \Omega_2, \Omega_3)^t$ respectively. It follows that the kinematic equations of a rigid body are given by: $\dot{b} = R\nu$, $\dot{R} = R\hat{\Omega}$ where the operator $\hat{*} : I\!R^3 \rightarrow so(3)$ is defined by $\hat{y}z = y \times z$ where $so(3)$ is the space of skew-symmetric matrices.

Assumption **2.1** *We take the origin of the body-fixed frame to be the center of gravity C_G. Moreover, we assume the body to have three planes of symmetry with body axes which coincide with the principal axes of inertia.*

Under our assumptions, the kinetic energy of the rigid body is given by

$$T_{body} = \frac{1}{2} \begin{pmatrix} v \\ \Omega \end{pmatrix}^t \begin{pmatrix} mI_3 & 0 \\ 0 & J \end{pmatrix} \begin{pmatrix} v \\ \Omega \end{pmatrix} \tag{1}$$

where m is the mass of the rigid body, I_3 is the 3×3-identity matrix and J is the body inertia matrix. The equations of motion for a rigid body are:

$$\begin{aligned} M\dot{\nu} &= M\nu \times \Omega \\ J\dot{\Omega} &= J\Omega \times \Omega + M\nu \times \nu \end{aligned} \tag{2}$$

where $M = mI_3$. We now assume that the body is submerged in a real fluid. By *real fluid* we mean an ideal fluid which is not inviscid. Note that we assume a real fluid to be irrotational, when in practice this is not the case. Our assumptions on the vehicle imply that the body inertia, added mass and moment of inertia matrices are all diagonal, and the added cross-terms are zero. It follows that the total kinetic energy of our rigid body in an unbounded real fluid is given by $T = \frac{1}{2}(\nu^t(mI_3 + M_f)\nu + \Omega^t(J_b + J_f)\Omega)$ where M_f, J_f are referred to as the added-mass and the added moment of inertia.

Assumption **2.2** *We assume the drag force $D_\nu(\nu)$ and drag momentum $D_\Omega(\Omega)$ matrices are both diagonal. The contribution of these forces is quadratic in the velocities; $D_\nu^{ii}(\nu) = C_D \rho A \mid \nu_i \mid \nu_i$ where C_D (sphere) is 1.2 for laminar flow and 0.2 for turbulent flow, ρ is the density of the fluid and A is the projected surface area of the object. With this assumption, the drag force and momentum are non-differentiable functions and theoretical analysis becomes difficult. To avoid difficulties, some restrict vehicle motion to a single direction, hence $\mid \nu_i \mid \nu_i = \nu_i^2$. We do not want to make this assumption because at least rotations are needed in both directions. Based on our test bed vehicle, our computations for the total drag force with respect to velocity suggests a cubic function with no quadratic or constant term as a good approximation. Thus, the contribution to the translational motions is given by $D_\nu(\nu) = diag(D_\nu^{i2}\nu_i^3 + D_\nu^{i1}\nu_i)$ and to the rotational motions by $D_\Omega(\Omega) = diag(D_\Omega^{i2}\Omega_i^3 + D_\nu^{i1}\Omega_i)$ where D_ν^{ij}, D_Ω^{ij} are constant coefficients.*

We also consider the restoring force and restoring moment. The only moment due to restoring forces is the righting moment $-r_B \times R^t \rho g \mathcal{V} k$ where r_B is the vector from C_G to the center of buoyancy C_B, ρ is the fluid density, g the acceleration of gravity, $\mathcal{V}$ the volume of fluid displaced by the rigid body and k the unit vector pointing in the direction of gravity.

Definition **2.3** *Under our assumptions, the equations of motion, in the body-fixed frame, for a controlled rigid body submerged in a real fluid are given by:*

$$\begin{aligned} M\dot{\nu} &= M\nu \times \Omega + D_\nu(\nu)\nu + R^t \rho g \mathcal{V} k + \varphi_\nu \\ J\dot{\Omega} &= J\Omega \times \Omega + M\nu \times \nu + D_\Omega(\Omega)\Omega - r_B \times R^t \rho g \mathcal{V} k + \tau_\Omega \end{aligned} \tag{3}$$

where M accounts for the mass and added mass coefficients, J accounts for the body moments of inertia and added moments of inertia coefficients. The matrices $D_\nu(\nu), D_\Omega(\Omega)$ represent the drag force and momentum. And, $\varphi_\nu = (\varphi_{\nu_1}, \varphi_{\nu_2}, \varphi_{\nu_3})^t$ and $\tau_\Omega = (\tau_{\Omega_1}, \tau_{\Omega_2}, \tau_{\Omega_3})^t$ account for the control forces.

Remark **2.4** In (3) we assume that we have three forces acting at C_G along the three body-fixed axes and three pure torques about these three axes. We will refer to these controls as the six degree-of-freedom (DOF) controls. This is not realistic from a practical point of view since underwater vehicle controls generally do not act at C_G. We assume the vehicle controls represent the action of thrusters on the vehicle. Other representations are valid; ours is driven by the test-bed AUV we use in experiments. The forces from these thrusters do not act at C_G and the torques are obtained from the moments created by the forces. For test-bed experimentation, we must compute the transformation between the six DOF controls and the real controls corresponding to the thrusters. We address such transformation for our test-bed vehicle later in this paper.

Together, the kinematic equations of a rigid body and (3) form a first-order affine control system on the tangent bundle $T\,\mathrm{SE}(3)$ which represents the second-order *forced affine-connection control system* on SE(3)

$$\nabla_{\gamma'}\gamma' = \begin{pmatrix} M^{-1}\Big(D_\nu(\nu)\nu + +R^t \rho g \mathcal{V} k + \varphi_\nu\Big) \\ J^{-1}\Big(D_\Omega(\Omega)\Omega - r_B \times R^t \rho g \mathcal{V} k + \tau_\Omega\Big) \end{pmatrix}. \tag{4}$$

where ∇ is the Levi-Civita affine-connection for the Riemannian metric induced by the kinetic energy T. In the absence of dissipative forces, Equation (4) represents a *left-invariant affine-connection control system* on the Lie group SE(3). It is true in general that a forced affine-connection control system on a manifold Q is equivalent to an affine control system on TQ. The equivalence is realized via the *geodesic spray* of an affine-connection and the *vertical lift* of tangent vectors to Q. We show this for the submerged rigid body. We introduce $\chi = (\eta, \nu, \Omega)$, and let $\chi_0 = \chi(0)$ and $\chi_T = \chi(T)$ be the initial and final states for our submerged rigid body. We denote the control $\gamma = (\varphi_\nu, \tau_\Omega)$. Equation (4) is equivalent to the following affine control system:

$$\dot{\chi}(t) = Y_0(\chi(t)) + \sum_{i=1}^{6} Y_i(t)\gamma_i(t) \tag{5}$$

where the drift Y_0 is given by

$$Y_0 = \begin{pmatrix} R\nu \\ \Theta\Omega \\ M^{-1}[M\nu \times \Omega + D_\nu(\nu)\nu + R^t \rho g \mathcal{V} k] \\ J^{-1}[J\Omega \times \Omega + M\nu \times \nu + D_\Omega(\Omega)\Omega - r_B \times R^t \rho g \mathcal{V} k] \end{pmatrix} \tag{6}$$

where Θ is given by Equations (10)-(12). The input vector fields are given by $Y_i = (0, 0, \mathbb{I}_i^{-1})^t$ with $\mathbb{I}_i^{-1}$ being the column i of the matrix $\mathbb{I}^{-1} = \begin{pmatrix} M^{-1} & 0 \\ 0 & J^{-1} \end{pmatrix}$. In other words, we have that $Y_i = \mathrm{vlft}(\mathbb{I}_i^{-1})$. In [5] the authors show that trajectories for the affine-connection control system on Q map bijectively to trajectories for the affine control system on TQ whose initial points lie on the zero-section. This concludes the general derivation of the equations of motion. We now show the local coordinates of the equations of motion for a rigid body submerged in a real fluid. We denote by $\eta = (b_1, b_2, b_3, \phi, \theta, \psi)^t$ the position and orientation of the vehicle with respect to the earth-fixed reference frame. The coordinates ϕ, θ, ψ are the Euler angles for the body frame. Translational and rotational velocities are $\nu = (\nu_1, \nu_2, \nu_3)^t$ and $\Omega = (\Omega_1, \Omega_2, \Omega_3)^t$.

LEMMA **2.5** *The equations of motion expressed in coordinates of the body-fixed frame for a rigid body submerged in a real fluid subjected to external forces are given by the following affine control system:*

$$\dot{b}_1 = \nu_1 \cos\psi \cos\theta + \nu_2 R^{12} + \nu_3 R^{13} \tag{7}$$

$$\dot{b}_2 = \nu_1 \sin\psi \cos\theta + \nu_2 R^{22} + \nu_3 R^{23} \tag{8}$$

$$\dot{b}_3 = -\nu_1 \sin\theta + \nu_2 \cos\theta \sin\phi + \nu_3 \cos\theta \cos\phi \tag{9}$$

$$\dot{\phi} = \Omega_1 + \Omega_2 \sin\phi \tan\theta + \Omega_3 \cos\phi \tan\theta \tag{10}$$

$$\dot{\theta} = \Omega_2 \cos\phi - \Omega_3 \sin\phi \tag{11}$$

$$\dot{\psi} = \frac{\sin\phi}{\cos\theta}\Omega_2 + \frac{\cos\phi}{\cos\theta}\Omega_3 \tag{12}$$

$$\dot{\nu}_1 = \frac{1}{m_1}[-(m_3)\nu_3\Omega_2 + (m_2)\nu_2\Omega_3 + D_\nu(\nu_1) - G\sin\theta + \varphi_{\nu_1}] \tag{13}$$

$$\dot{\nu}_2 = \frac{1}{m_2}[(m_3)\nu_3\Omega_1 - (m_1)\nu_1\Omega_3 + D_\nu(\nu_2) + G\cos\theta\sin\phi + \varphi_{\nu_2}] \tag{14}$$

$$\dot{\nu}_3 = \frac{1}{m_3}[-(m_2)\nu_2\Omega_1 + (m_1)\nu_1\Omega_2 + D_\nu(\nu_3) + G\cos\theta\cos\phi + \varphi_{\nu_3}] \tag{15}$$

$$\begin{aligned}\dot{\Omega}_1 = \frac{1}{I_{b_1} + J_f^{\Omega_1}}&[(I_{b_2} - I_{b_3} + J_f^{\Omega_2} - J_f^{\Omega_3})\Omega_2\Omega_3 + (M_f^{\nu_2} - M_f^{\nu_3})\nu_2\nu_3 \\ &+ D_\Omega(\Omega_1) + \rho g\mathcal{V}(-y_B\cos\theta\cos\phi + z_B\cos\theta\sin\phi) + \tau_{\Omega_1}]\end{aligned} \tag{16}$$

$$\begin{aligned}\dot{\Omega}_2 = \frac{1}{I_{b_2} + J_f^{\Omega_2}}&[(I_{b_3} - I_{b_1} + J_f^{\Omega_3} - J_f^{\Omega_1})\Omega_1\Omega_3 + (M_f^{\nu_3} - M_f^{\nu_1})\nu_1\nu_3 \\ &+ D_\Omega(\Omega_2) + \rho g\mathcal{V}(z_B\sin\theta + x_B\cos\theta\cos\phi) + \tau_{\Omega_2}]\end{aligned} \tag{17}$$

$$\begin{aligned}\dot{\Omega}_3 = \frac{1}{I_{b_3} + J_f^{\Omega_3}}&[(I_{b_1} - I_{b_2} + J_f^{\Omega_1} - J_f^{\Omega_2})\Omega_1\Omega_2 + (M_f^{\nu_1} - M_f^{\nu_2})\nu_1\nu_2 \\ &+ D_\Omega(\Omega_3) + \rho g\mathcal{V}(-x_B\cos\theta\sin\phi - y_B\sin\theta) + \tau_{\Omega_3}]\end{aligned} \tag{18}$$

where $G = mg - \rho g\mathcal{V}$, $m_i = m + M_f^{\nu_i}$, $D_\nu(\nu_i) = D_\nu^{i2}\nu_i^3 + D_\nu^{i1}\nu_i$ *and* $D_\Omega(\Omega_i) = D_\Omega^{i2}\Omega_i^3 + D_\Omega^{i1}\Omega_i$. $\varphi_\nu = (\varphi_{\nu_1}, \varphi_{\nu_2}, \varphi_{\nu_3})$ *and* $\tau_\Omega = (\tau_{\Omega_1}, \tau_{\Omega_2}, \tau_{\Omega_3})$ *represent the control.*

Definition **2.6** *An admissible control is a measurable bounded function* $(\varphi_\nu, \tau_\Omega) : [0, T] \to U = F \times \mathcal{T}$ *where:*

$$\begin{aligned}\mathcal{F} &= \{\varphi_\nu \in \mathbb{R}^3 | \alpha_{\nu_i}^{\min} \leq \varphi_{\nu_i} \leq \alpha_{\nu_i}^{\max},\ \alpha_{\nu_i}^{\min} < 0 < \alpha_{\nu_i}^{\max},\ i = 1, 2, 3\} \\ \mathcal{T} &= \{\tau_\Omega \in \mathbb{R}^3 | \alpha_{\Omega_i}^{\min} \leq \tau_{\Omega_i} \leq \alpha_{\Omega_i}^{\max},\ \alpha_{\Omega_i}^{\min} < 0 < \alpha_{\Omega_i}^{\max},\ i = 1, 2, 3\}\end{aligned} \tag{19}$$

3 Maximum Principle

Before describing our algorithm, we recall the necessary conditions of the maximum principle. This is to introduce terminology used in our explanations, namely the notion of bang-bang and singular arcs. We state the maximum principle without making use of the geometric structure of our problem since we use the equations of motion expressed in local coordinates. We do this to introduce a vocabulary and not conduct an analysis of the solutions. We refer the interested reader to [6] for a geometric study of the extremals. Assume that there exists an admissible time-optimal control $\gamma = (\varphi_\nu, \tau_\Omega) : [0,T] \to \mathcal{U}$, such that the corresponding trajectory $\chi = (\eta, \nu, \Omega)$ is a solution of the equations (7)-(18) and steers the body from χ_0 to χ_T. The maximum principle [2], implies that there exists an absolutely continuous vector $\lambda = (\lambda_\eta, \lambda_\nu, \lambda_\Omega) : [0,T] \to I\!R^{12}$, $\lambda(t) \neq 0$ for all t, such that the following conditions hold almost everywhere:

$$\dot{\eta} = \frac{\partial H}{\partial \lambda_\eta},\ \dot{\nu} = \frac{\partial H}{\partial \lambda_\nu},\ \dot{\Omega} = \frac{\partial H}{\partial \lambda_\Omega},\quad \dot{\lambda}_\eta = -\frac{\partial H}{\partial \eta},\ \dot{\lambda}_\nu = -\frac{\partial H}{\partial \nu},\ \dot{\lambda}_\Omega = -\frac{\partial H}{\partial \Omega} \tag{20}$$

where the Hamiltonian function H is given by:

$$\begin{aligned} H(\chi, \lambda, \varphi, \tau) = &\lambda_\eta^t (R\nu, \Theta\Omega)^t + \lambda_\nu^t M^{-1}[M\nu \times \Omega + D_\nu(\nu)\nu + R^t \rho g \mathcal{V} k + \varphi_\nu] \\ &+ \lambda_\Omega^t J^{-1}[J\Omega \times \Omega + M\nu \times \nu + D_\Omega(\Omega)\Omega - r_B \times R^t \rho g \mathcal{V} k \\ &+ \tau_\Omega] \end{aligned} \tag{21}$$

Furthermore, the maximum condition holds:

$$H(\chi(t), \lambda(t), \varphi_\nu(t), \tau_\Omega(t)) = \max_{\gamma \in \mathcal{U}} H(\chi(t), \lambda(t), \gamma_\nu, \gamma_\Omega) \tag{22}$$

The maximum of the Hamiltonian is constant along the solutions of (20) and must satisfy $H(\chi(t), \lambda(t), \varphi_\nu(t), \tau_\Omega(t)) = \lambda_0$, $\lambda_0 \geq 0$. A quadruple $(\chi, \lambda, \varphi_\nu, \tau_\Omega)$ that satisfies the maximum principle is called an extremal, and the vector function $\lambda(.)$ is called the adjoint vector. The maximum condition (22), along with the control domain $\mathcal{F} \times \mathcal{T}$, is equivalent almost everywhere to (M, J diagonal and > 0), $i = 1, 2, 3$:

$$\varphi_{\nu_i}(t) = \alpha_{\nu_i}^{\min} \text{ if } \lambda_{\nu_i}(t) < 0 \quad \text{and} \quad \varphi_{\nu_i}(t) = \alpha_{\nu_i}^{\max} \text{ if } \lambda_{\nu_i}(t) > 0 \tag{23}$$

$$\tau_{\Omega_i}(t) = \alpha_{\Omega_i}^{\min} \text{ if } \lambda_{\Omega_i}(t) < 0 \quad \text{and} \quad \tau_{\Omega_i}(t) = \alpha_{\Omega_i}^{\max} \text{ if } \lambda_{\Omega_i}(t) > 0 \tag{24}$$

Clearly, the zeros of the functions λ_{ν_i}, λ_{Ω_i} determine the structure of the solutions to the maximum principle, and hence of the time-optimal control.

Definition **3.1** *We say that a component of the control is bang-bang on a given interval $[t_1, t_2]$ if its corresponding switching function is nonzero for almost all $t \in [t_1, t_2]$. The bang-bang component of the control only takes values in $\{\alpha_i^{\min}, \alpha_i^{\max}\}$ for almost every $t \in [t_1, t_2]$, $i = 1, \cdots, 6$.*

Definition **3.2** *If there is a nontrivial interval $[t_1, t_2]$ such that a switching function is identically zero, the corresponding component of the control is said to be singular on $[t_1, t_2]$. A singular control is said to be strict if the other controls are bang.*

Below is the notion of a switching time as used in this paper.

DEFINITION **3.3** *We consider two types of switching times. First, assume a given component of the control to be piecewise constant, in particular it is the case if this component is bang-bang. Then, we say that* $t_s \in [t_1, t_2]$ *is a switching time for this component if, for each interval of the form* $]t_s - \varepsilon, t_s + \varepsilon[\cap [t_1, t_2]$, $\varepsilon > 0$, *the component is not constant. Secondly, a time* t_s *will also be referred to a switching time for a given component if it corresponds to a junction between a singular and a bang-bang arc for this component. We do not consider the case of a chattering junction between bang and singular case. For this issue, the interested reader should refer to [8]. Finally, when computing the total number of switching times along a trajectory we count only 1 switching time in the case that several components of the control switch simultaneously.*

4 Numerical Algorithms

We are now ready to introduce the numerical methods used for our computations and analyze the obtained results.

4.1 Optimization Method

To numerically solve an optimal control problem (OCP) we have two broad classes of methods: indirect or direct. Indirect methods are based on the application of the maximum principle and are usually called single or multiple shooting methods. The single shooting method consists of computing extremals of the (OCP). The idea is based on the existence of a control feedback $\gamma(\chi, \lambda)$ in terms of the state and adjoint variables, the feedback is provided by the maximization condition (22). Given an initial value $\lambda(0)$ of the adjoint vector and a final time T, we integrate the Hamiltonian system (20) using the feedback control previously determined. This then becomes an initial value problem. The results are the final values for the state and the adjoint variables: $\chi(T)$ and $\lambda(T)$. If $\chi(T) = \chi_T$, then we have found an extremal of our problem. If not, we search for a zero of $\chi(T) - \chi_T$ using a Newton-like algorithm applied to $\lambda(0)$ and T. The function $S(T, \lambda(0)) = \chi(T) - \chi_T$ is called the shooting function. For bang-bang controls, S is not differentiable everywhere and especially not for pairs $(T, \lambda(0))$ which generate a new switching time (the structure of the control is not fixed in neighborhoods of the pair). Direct methods are a rewriting of the (OCP) as a finite dimensional optimization problem. There are many ways to rewrite the (OCP), however we only state the one used for our computations. We reparameterize the time domain $[0, T]$ as $[0, 1]$ and choose a discretization $0 = t_0 < t_1 < \cdots < t_N = 1$ of $[0, 1]$. Then

we write the discretized (OCP) with unknowns T, $\chi_i = \chi(t_i),\ i = 1, \cdots, N$ and $\gamma_i,\ i = 0, \cdots, N-1$. The result is a large-scale nonlinear optimization problem whose nonlinear constraints are the discretized dynamics of the form (for an Euler scheme) $\chi_{i+1} = \chi_i + (t_{i+1} - t_i)\dot{\chi}_i(\chi_i, \gamma_i)$, $i = 0, \cdots, N-1$ and $\chi_N = \chi_T$. We call this non-linear problem (NLP).
Let us now compare the methods. The Newton algorithm within the indirect method is known to be sensitive to initialization. Hence it is nearly impossible to find a zero of S without any *a priori* knowledge of the structure of the optimal control or without a clever initialization process. Singular arcs especially are quite difficult to integrate and to locate. The advantage of a direct method is the robustness with respect to the initialization. Also it is easy to add state constraints to the original (OCP). The disadvantage is that direct methods are computationally very demanding since a discretization of (OCP) usually yields a large number of parameters to optimize (we need N large enough so that the discretization makes sense with respect to the continuous (OCP)). Having not yet found a clever Newton initialization process, we base our computations on direct methods which have yielded good results.
Methods to solve nonlinear optimization problems are well developed. We choose to use the interior point method *IpOpt*, see [25], together with the modeling language *AMPL*, see [14]. For our direct method, we use Heun's fixed step integration scheme. Moreover, we have additional constraints on the final state and the upper and lower bounds of the controls.
First consider the situation in which the initial configuration is the origin and we want to reach $\eta_T = (6, 4, 1, 0, 0, 0)$ [3] with both configurations being at rest. Figure 1 shows the time optimal strategy with the control bounds given in Section 5. The time for this trajectory is $t^f_{NLP_N} \approx 25.85$s. Let us compare this strategy to other strategies linking these same configurations.

DEFINITION **4.1** *A pure translation in the body fixed-frame is a motion along one of the body fixed-frame axes. We have three pure translations in the body fixed frame: a pure surge, a pure sway and a pure heave. Similarly, we define the pure rotations in the body fixed-frame corresponding to motions resulting from the action of a pure torque around one of the body fixed-frame axes. We have a pure roll, pure pitch and pure yaw.*

Pure motions are very natural to consider since we can join any two configurations through a concatenation of at most six pure motions. In Figure 2, we show such a concatenated pure motion strategy; displaying only graphs of variables which are not identically zero. Note that this trajectory is formed by a pure surge acceleration during $t^{acc}_{surge} \approx 38.39$ s, a deceleration for $t^{dec}_{surge} \approx 3.74$ s, a pure sway acceleration for $t^{acc}_{sway} \approx 25.89$ s, a deceleration for $t^{dec}_{sway} \approx 3.74$ s, a pure heave acceleration for $t^{acc}_{heave} \approx 2.92$ s and a deceleration for $t^{dec}_{heave} \approx 5.24$

[3] This final configuration is chosen to maximize the viewable area as seen from the video camera recording our experiments.

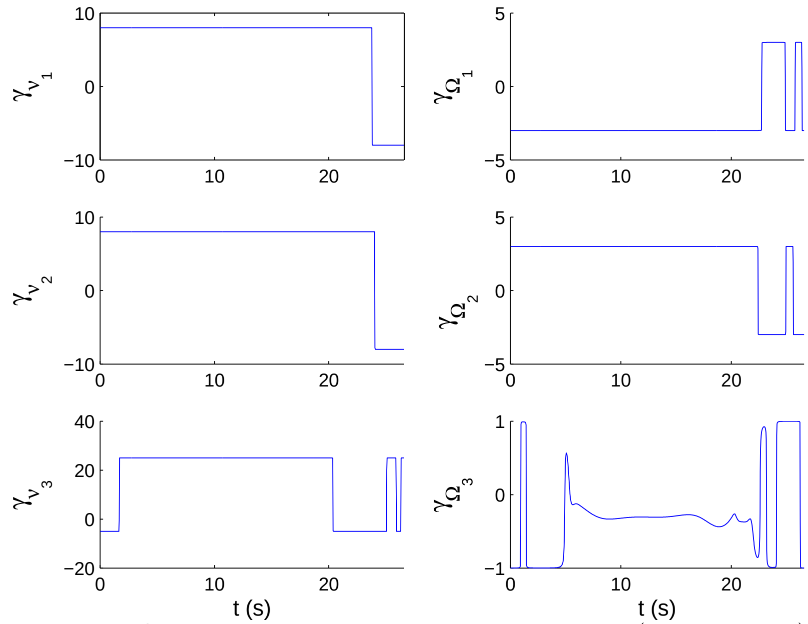

Fig. 1. *Time Optimal thrust strategy ending at* $\eta_T = (6, 4, 1, 0, 0, 0)$ *at rest.*

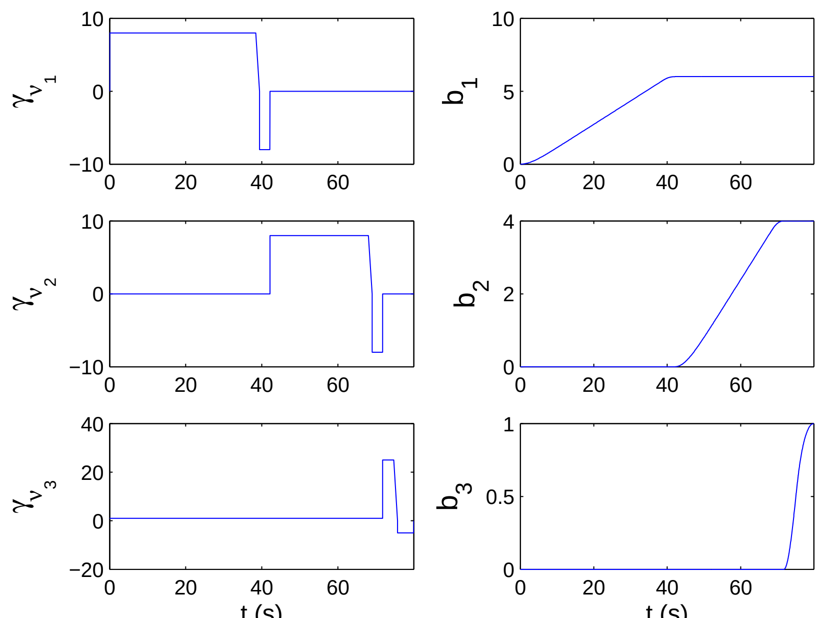

Fig. 2. *Pure surge, sway and then heave in order to reach* $\eta_T = (6, 4, 1, 0, 0, 0)$ *m.*

s. The non-symmetry of the acceleration and deceleration phases is due to drag forces and thruster unsymmetries. The total transfer time for this trajectory is $t_{pure} \approx 79.92$ s. The duration is more than triple the optimal time! This is actually not that surprising since the pure motion trajectory uses only a fraction of the available thrust. A pure motion control strategy is attractive for our problem due to its piecewise constant structure but it is far from time efficient. Notice that the same should be true when considering energy consumption. It is inefficient to concatenate motions through configurations at rest. Now back to the time optimal strategy. The structure here is mostly bang-bang, except for the τ_{Ω_3} control which contains singular arcs. These singular arcs depend on our choice of initial and final configurations. Orienting the vehicle correctly allows it to use the full power of the translational controls, but it needs to maintain this orientation over the entire trajectory. Singular arcs do not appear in τ_{Ω_1} and τ_{Ω_2} because their full power is needed to offset the restoring moments. The translational controls $\varphi_{\nu_1,\nu_2,\nu_3}$ are used to their full extent, as one would expect for a time optimal translational displacement. Other than the τ_{Ω_3} singular arc, the time optimal control strategy has another severe drawback when considering practical implementation. From Figure 1, we can count 21 actuator switching times. Each switching generates errors since the physical thrusters have an unstable behavior for abrupt changes of direction. We cannot match the accuracy of the solution with the refresh rate on a real vessel because it would require storage of impractical amounts of data. From a computational point of view, obtaining the time optimal trajectory is time consuming and data storage limiting. To accurately track the singular arcs and handle the large number of switches, one solution similar to that represented in Figure 1 requires about 15 minutes of computation.

From the above remarks, neither a control strategy based on pure motions nor one based on time optimal trajectories alone is a viable option for practical implementation onto a test-bed vehicle. The next section takes the advantages of both control strategies and combines them to create the new hybrid method.

4.2 Switching time parameterization algorithm

Inspired from the work in [15,23], we developed another approach to overcome the issues seen in the previous solving methods. In [23], the authors use the discretized solution of an optimal control problem to extract the switching structure of the optimal control. Then, they rewrite the optimal control problem as a nonlinear optimization problem whose unknowns are the switching times (more precisely the time length between two switches). A high order integrator can be used to integrate the obtained dynamic system. The motivation for their approach is the verification of second order sufficient conditions for optimality. Along singular trajectories they are allowed to write the control as a feedback both the state and adjoint variables. This can not be done in our situation. In [6] we detail the computations for the singular components of the control. The formulas for the singular components are very complex and the feedback depends on the adjoint variables. Moreover, the structure of the optimal solution is very difficult, if not impossible, to extract.

It is important to note that our primary goal is to produce a time efficient which can be easily implemented onto a test-bed vehicle. By time efficient, we mean that the trajectory duration is close in time to that of the time optimal solution. To this end, we first note that a translational displacement can always be achieved by a thrust strategy with a single switching time at which the 3 components of φ_ν change. We extend this idea by imposing the structure of the control strategy but not basing it on the solution of the (NLP) problem. We fix the number of switching times along the trajectory, preferably to a small number, and we numerically determine the optimal trajectory from these candidates. We call this new optimization problem $(STPP)_p$ (Switching Time Parameterization Problem) where p refers to the number of switching times. The unknowns are the time periods between two switching times along with the time period between the last switch and the final time, and the values of the constant thrust arcs. It is essential for convergence that the later values are introduced as parameters. Our construction does not necessarily produce bang-bang trajectories. The new optimization problem $(STPP)_p$ has the following form:

$$(STPP)_p \begin{cases} \underset{z\in\mathcal{D}}{min} \;\; t_{p+1} \\ t_0 \quad\;\; = 0 \\ t_{i+1} \;\; = t_i + \xi_i \;, \;\; i = 1, \cdots, p \\ \chi_{i+1} = \chi_i + \int_{t_i}^{t_{i+1}} \dot{\chi}(t, \gamma_i) dt \\ \chi_{p+1} = \chi_T \\ z \quad\;\; = (\xi_1, \cdots, \xi_{p+1}, \gamma_1, \cdots, \gamma_{p+1}) \\ \mathcal{D} \quad\;\; = I\!R_+^{(p+1)} \times \mathcal{U}^{p+1} \end{cases} \tag{25}$$

where $\xi_i,\; i = 1, \cdots, p+1$ are the time arclengths and $\gamma_i \in \mathcal{U},\; i = 1, \cdots, p+1$ are the values of the constant thrust arcs. The right hand side of the dynamic system defined by (7)-(18) is simply $\dot{\chi}(t, \gamma_i)$, with the constant control γ_i. To integrate the dynamic system of $(STPP)_p$ we use *DOP853*, a high order adaptative step integrator [16]. This allows us to minimize potential differences between the theoretical computations and the experiments. Uncertainties are already introduced with the approximated model. Compared to the computation time of the solutions of (NLP), we gain considerable computational time with our $(STPP)_p$ procedure. The reason is the drastic reduction in the number of unknowns. Even though the integration takes more time, the computational resources needed to solve $(STPP)_p$ are drastically reduced. Note that $(STPP)_p$ is another way to discretize the (OCP) which could be called a variable step recursive discretization. *Variable step* implies the arclength between two discretization times is an optimization parameter. *Recursive* implies that the control is an optimization parameter and the final value of the state is computed recursively from the initial to the final time. If N is large enough (to insure convergence for (NLP)), the solutions of $(STPP)_{N-1}$ are better than the solutions of $(NLP)_N$. Clearly, the following inequalities are true:

$$t^f_{\min} \leq t^f_{STPP_{p+1}} \leq t^f_{STPP_p} \leq t^f_{NLP_p} \tag{26}$$

where $t^f_{NLP_p}$ is the time of the solution of (NLP) with p discretization points, and $t^f_{\min}$ is the theoretical minimum time. However, we are interested in solution of $(STPP)_p$ when p is small.

In Figures 3 and 4, we compare the solution of (NLP) for the initial and final configurations used in Section 4.1 with the solution of $(STPP)_4$ for the same set of configurations. Notice that the final time $t^f_{STPP_4}$ is less than 6% more

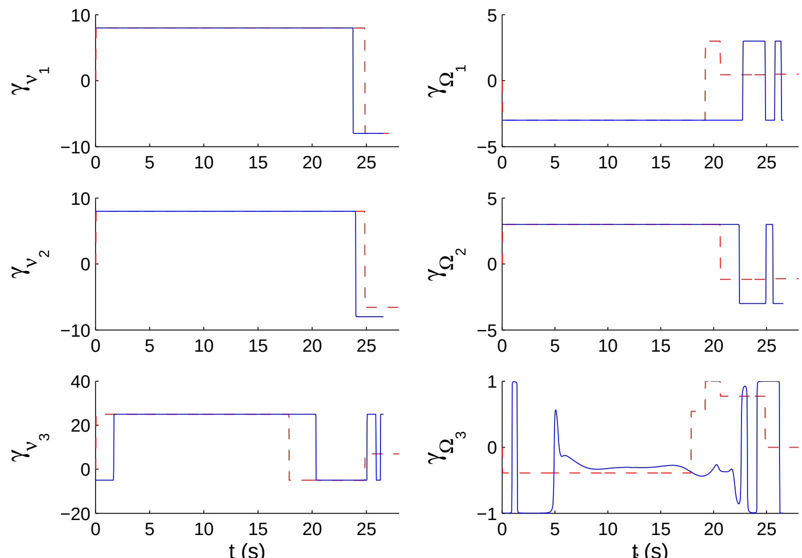

Fig. 3. *Control strategy solution of* (NLP) *(*$N = 1000$*,* $t^f_{NLP} \approx 26.58$ *s, solid line) and of* $(STPP)_4$ *(*$t^f_{STPP_4} \approx 28.02$ *s, dashed line).*

than the time optimal trajectory. Through many experiments, we have shown that the trajectory time computed using the switching times parameterization algorithm is within 10% of the optimal trajectory time. An open question is a formal proof of this result. With our new algorithm we produce not only trajectories that are easily implementable and time efficient but we also dramatically reduced the computational time. Solving $(STPP)_p$ takes less than 30 s on the same platform which earlier quoted 15 minutes to solve (NLP). The simulations show a sensitivity in the initialization process of our switching time parameterization algorithm and that there are few local minima. The STPP strategy is easier to implement than the (NLP) because of the reduced number of switching times and the piecewise constant property. As seen in the graphs, the control strategies of (NLP) and $(STPP)_4$ share some properties. This suggests the use of $(STPP)_p$ solutions as an initialization point to the (NLP). However, simulations have shown that the time gained using such a procedure is not substantial, and we still have the problem of the singular arcs.

From the two control strategies seen in Figure 3, we display the corresponding theoretical trajectories in Figure 4. The two trajectories are very similar. For

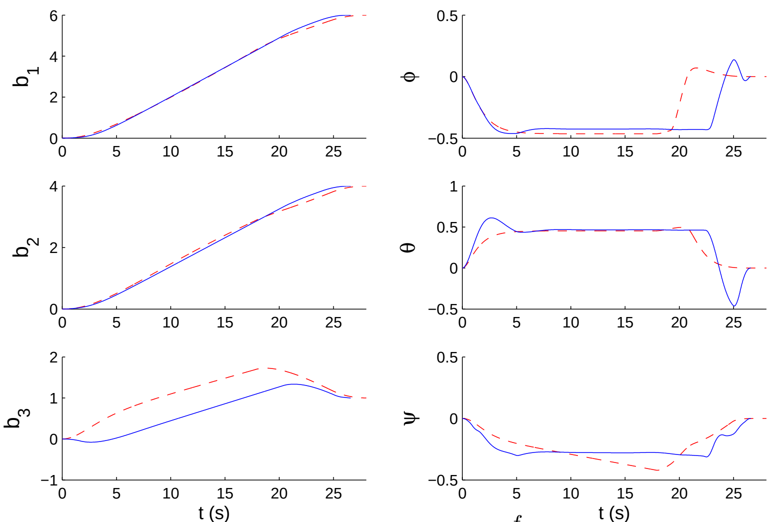

Fig. 4. *Trajectory solution of* (NLP) *(*$N = 1000$*,* $t^f_{NLP} \approx 26.58$ *s, solid line) and of* $(STPP)_4$ *(*$t^f_{STPP_4} \approx 28.02$ *s, dashed line).*

the optimal trajectory (solid line), the evolution of ψ, during the singular arc, is constant. Whereas for the $(STPP)_4$ solution (dashed line), ψ is slightly evolving over the entire trajectory. In Table 1 we analyze $t^f_{STPP_p}$ for different p and alternate final configurations. The initial configuration is always $\eta_0 = 0$ and $\nu_0 = \Omega_0 = 0$. Because of instant control (thruster) switches, the reader

Final Configuration	(NLP)			$(STPP)_p$		
	t^f_{NLP}	# sw.	Singular?	$t^f_{STPP_2}$	$t^f_{STPP_3}$	$t^f_{STPP_4}$
$(6, 4, 1, 0, 0, 0)$	26.58 s	21	Yes	28.72 s	28.10 s	28.02 s
$(6, 4, 0, 0, 0, 0)$	28.42 s	28	Yes	34.43 s	29.83 s	29.01 s
$(6, 0, 0, 0, 0, 0)$	25.40 s	23	Yes	31.52 s	28.45 s	26.55 s
$(0, 6, 0, 0.2, 0.3, 0)$	25.46 s	19	Yes	30.09 s	29.05 s	28.98 s

Table 1

Final times for different final configurations (#sw. = number of switchings)

may be surprised that piecewise constant control strategies are easily implementable on a test-bed vehicle. We remedied this by connecting the piecewise constant arcs by a linear function which ramps the control from max to min or vise versa. This linear function is chosen in such a way that its slope does not damage the AUV's electronics (i.e. vehicle specific). The vehicle used in the experiments has an internal refresh rate $R = 33$ Hz. Actuator switchings generally occur over a period of a few R intervals. We also implemented electronic safety circuits to control any induced voltage or current.

5 Experiments

5.1 Test-Bed AUV

The test-bed AUV we use is the Omni-Directional Intelligent Navigator (ODIN) which is owned and operated by the Autonomous Systems Laboratory (ASL), College of Engineering at the University of Hawaii. The experiments conducted for this research are carried out at the Duke Kahanamoku Swimming Complex at the University of Hawaii. As seen in Figure 5, ODIN has a spherical

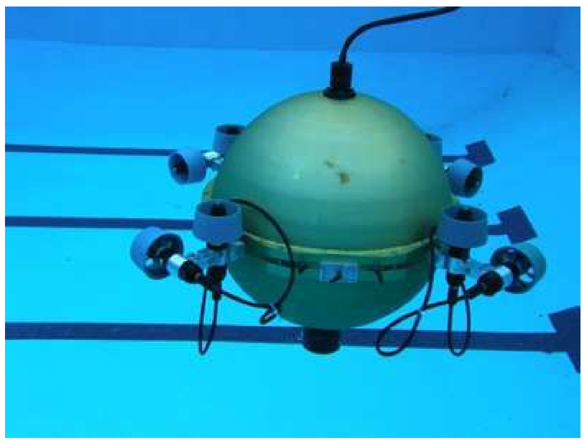

Fig. 5. *ODIN operating in the pool.*

hull which is 65cm in diameter. This sphere is constructed from an aluminum alloy to prevent corrosion. Eight thrusters are attached to the sphere via four fabricated mounts, each holding two thrusters. The thrusters are evenly distributed around the sphere with four vertical and four horizontal. Fully assembled, ODIN weighs $126.55kg$ and is positively buoyant by $\approx 0.1kg$. ODIN is capable of moving in six DOF from either a remote or autonomous mode. For our experiments, ODIN is tethered, but only to send commands via TCP/IP protocol from a shore based laptop to be run in autonomous mode. This setup allows for multiple tests to be conducted without removing ODIN from the water to upload mission sorties. ODIN's internal CPU is a 800 MHz Intel based processor running on a PC104+ form factor with two external I/O boards providing A/D and D/A operations. Major internal components include a pressure sensor, inertial measurement unit (IMU) and 24 batteries. ODIN is capable of computing real time, yaw, pitch, tilt, and depth and can run autonomously for up to 5 hours. However the IMU is not designed to track fast changes of heading ($\geq 6°/s$). The software is divided into two components. The first component is based on a real time extension to the Windows 2000 operating system, which provides ODIN real time autonomous control. The second component runs on the remote laptop and allows the operator to upload autonomous mission profiles to ODIN on the fly during testing as well as monitor ODIN in real time. As noted above, ODIN does not have real time sensors to detect horizontal (x, y) position. Instead, experiments are video taped from the 10m diving platform, giving us a near nadir view of ODIN's movements. Videos are saved and horizontal position is post processed for later analysis. A real time system utilizing sonar was available on ODIN, but was abandoned for two main reasons. First, the sonar created too much noise in the diving well which led to inaccuracies. More significantly, in the implementation of our efficient trajectories, ODIN is often required to achieve large ($> 15°$) list angles which render the sonars useless for horizontal position. Many solutions were attempted and video led to the most accurate results. The numerical values of the various parameters used for our theoretical model are given in Table 2. These values were derived from experiments performed on ODIN. The added mass and drag terms were estimated from formulas found in [19,21]. Moments of inertia were calculated using experiments outlined in [20]. We used inclining experiments to locate and place C_B while we assume that C_G is located at the center of our body-fixed axis. The drag and C_B estimates were then adapted to match the experimental behavior of the vehicle. Unique to ODIN's

m	126.55 kg	$\rho g \nabla$	1241.2 N	C_B	$(0.9, 0.2, -7)$ mm
M_f^u	70 kg	M_f^v	70 kg	M_f^w	70 kg
I_x	5.46 kg.m^2	I_y	5.29 kg.m^2	I_z	5.72 kg.m^2
J_f^p	0 kg.m^2	J_f^q	0 kg.m^2	J_f^r	0 kg.m^2
D_ν^{11}	-27.03	D_ν^{21}	-27.03	D_ν^{31}	-27.03
D_ν^{12}	-897.66	D_ν^{22}	-897.66	D_ν^{32}	-897.66
D_Ω^{11}	-13.79	D_Ω^{21}	-13.79	D_Ω^{31}	-11.94
D_Ω^{12}	-6.46	D_Ω^{22}	-6.46	D_Ω^{32}	-6.94

Table 2
Numerical values used for our hydrodynamic model.

construction is the control from an eight dimensional thrust to move in six DOF. This construction puts redundancy into the system in case of thruster failure. It is important to distinguish between a control for the real vehicle, namely the applied control referring to the action of the thrusters, and the six DOF control introduced previously. Our input trajectories to ODIN take the form of the six DOF controls which are converted onboard ODIN to the control for the eight actual thrusters using the following Thrust Control Matrices (TCM's) (Eqns. 27 and 28).

$$TCM\ horizontal = \begin{bmatrix} -0.707 & 0.707 & 0.707 & -0.707 \\ 0.707 & 0.707 & -0.707 & -0.707 \\ 0.48160 & -0.48160 & 0.48160 & -0.48160 \end{bmatrix} \quad (27)$$

$$TCM\ vertical = \begin{bmatrix} -1.0 & -1.0 & -1.0 & -1.0 \\ -0.26989 & -0.26989 & 0.26989 & 0.26989 \\ 0.26989 & -0.26989 & -0.26989 & 0.26989 \end{bmatrix} \qquad (28)$$

These transformations are based on the following assumptions. Let us denote $\gamma_i^h,\ i = 1, \cdots, 4$ as the thrusts induced by the horizontal thrusters and $\gamma_i^v,\ i = 1, \cdots, 4$ the thrusts induced by the vertical thrusters. The first assumption is that points of application of the thrusts $\gamma_i^{(h,v)}$ lie in a plane going through the center of the vehicle. We also assume that the distance from the center of the body frame (C_G in our case) to the center of the sphere (C_B in our case) is small with respect to the radius of the sphere. As a consequence we can decouple the action of the thrusters as follows. The horizontal thrusters contribute only to the forces φ_{ν_1} (surge) and φ_{ν_2} (sway) and to the torque τ_{Ω_3} (yaw). The vertical thrusters contribute only to the force φ_{Ω_3} (heave) and to the torques τ_{ν_1} (roll) and τ_{ν_2} (pitch). We have $(\varphi_{\nu_1}, \varphi_{\nu_2}, \tau_{\Omega_3})^t = TCM\ horizontal \cdot (\gamma^h)^t$ and $(\varphi_{\nu_3}, \tau_{\Omega_1}, \tau_{\Omega_2})^t = TCM\ vertical \cdot (\gamma^v)^t$. Assuming that the thrusters are independently powered, we can reasonably state that each thrust $\gamma_i^{(h,v)}$ is bounded by fixed values:

$$\gamma^{(h,v)} \in \Upsilon = \{\gamma \in I\!R^8 | \gamma_i^{(h,v),\min} \leq \gamma_i^{(h,v)} \leq \gamma_i^{(h,v),\max}, i = 1, \cdots, 4\} \qquad (29)$$

The image of Υ through the above linear transformation is composed of two flat ellipsoids. We choose a box included within these ellipsoids as domain of control for the six DOF control. There are different possible choices for this box depending on the controllability properties that we prefer for our vehicle. In the sequel, we assume the control domain for the component φ_ν and τ_Ω to be as in Equation (19). For our numerical computations, we will take $\alpha_{\nu_{1,2}}^{\max} = -\alpha_{\nu_{1,2}}^{\min} = 8$ N, $\alpha_{\nu_3}^{\max} = 25$ N, $\alpha_{\nu_3}^{\min} = -5$ N, $\alpha_{\Omega_{1,2}}^{\max} = -\alpha_{\Omega_{1,2}}^{\min} = 3$ N.m and $\alpha_{\Omega_3}^{\max} = -\alpha_{\Omega_3}^{\min} = 1$ N.m. The non-symmetry of $\alpha_{\nu_3}^{\min,\max}$ is due to the fact that the 4 vertical thrusters are all oriented in the same direction. Along with the tests to determine the values in Table 2, we also tested the thrusters. Each thruster has a unique voltage input to power output relationship. This relationship is highly nonlinear and is approximated using a piecewise linear function which we refer to as our thruster model.

5.2 One Switching STTP Experiment

We begin by testing a $(STPP)_1$ strategy from the origin to $\eta_T = (6, 4, 1, 0, 0, 0)$. This control strategy contains only one switch and thus does not change the orientation of the vehicle during the motion. This aids the practical implementation since we can apply roll, pitch and yaw feedback stabilization during the experiment. The only open loop controls are the translational φ_{ν_1}, φ_{ν_2} and φ_{ν_3}. Here the linear function connecting the constant thrust arcs occur over a duration of 0.9 s. This linear function minimally increases the total time when compared to the purely piecewise constant $(STPP)$ strategy. Figure 6 displays the applied translational controls. The evolution of ODIN's position during the experiment is given by the solid line and the prescribed evolution (theoretical evolution we wanted to achieve) is given by the dashed line. Note that, from the point of view of this paper, there is no need to display the orientation evolution or the angular control since they are computed in closed-loop with feedback and we are not interested in the efficiency of the feedback controller. The total time of this strategy is $t^f_{STPP_1} \approx 42.55$ s which is actually the

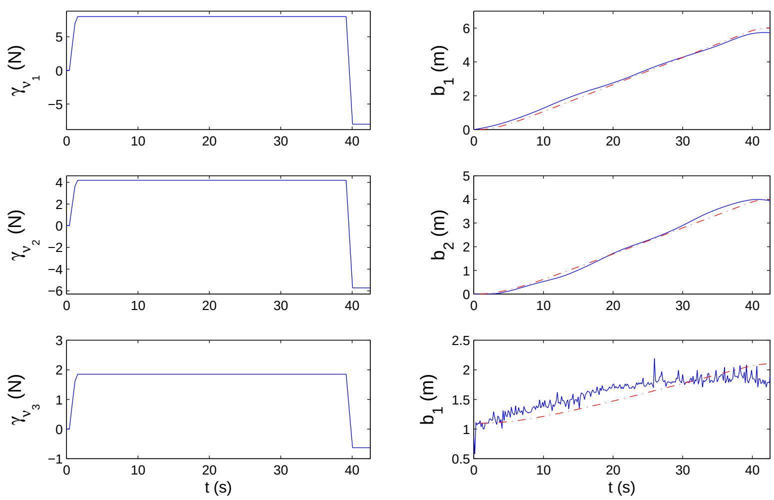

Fig. 6. *Experimental (solid) and prescribed (dashed) evolutions of the* AUV *for a* $(STPP)_1$ *strategy ending at* $\eta_T = (6, 4, 1, 0, 0, 0)$.

time needed to do just a pure surge motion of 6 m with the prescribed linear junction. This duration is not surprising at all since the $(STPP)_1$ method is simply a modified pure motion. Here the direction of the motion is not along one of the body fixed axes, but along a line in the direction from the initial to the final configuration. A $(STPP)_1$ strategy takes as long as the longest pure motion of a pure motion strategy. The evolutions of b_1 and b_2 exactly correspond to surge and sway since there is no orientation alteration. We see that the experimental is very close to the prescribed path. This indicates that our drag and thrust magnitudes are well estimated. We found similar behavior over various thrust magnitudes implying that our thrust and drag coefficient estimates are consistent. The behavior of b_3 (heave) is slightly more erratic and deviant than b_1 and b_2. This behavior is a result of noise within the depth sensor circuitry and buoyancy effects from the tether. Since ODIN is near neutrally buoyant, small buoyancy alterations have noticeable effects. This experiment validates part of our hydrodynamic model and thruster calibration. This gives very promising results even though the $(STPP)_1$ strategy is not exceptionally time efficient.

5.3 Two Switchings STPP Experiment

Building on the good results from Section 5.2, we now consider a $(STPP)_2$ strategy with the same final configuration. This trajectory contains a large list angle. This makes orientation tracking nearly impossible due to the poor dynamic behavior of our sensors (especially for ψ). Including the linear junction, the total duration of this trajectory is $t^f_{STPP_2} \approx 28.66$ s. This experiment does not use any feedback control. The results test and help update the accuracy of our hydrodynamic model as well as our thruster calibration. In Figure 7, we show the control strategy applied during the experiment. The $(STTP)_1$ strat-

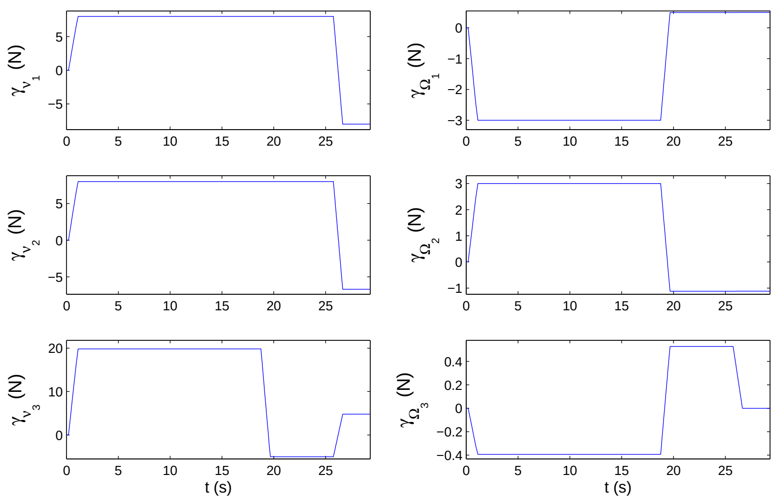

Fig. 7. $(STPP)_2$ *control strategy targeting* $\eta_T = (6, 4, 1, 0, 0, 0)$.

egy maximized only one control whereas here, the $(STPP)_2$ strategy used four of the controls to their maximum extent (γ_{ν_1}, γ_{ν_2}, γ_{Ω_1} and γ_{Ω_2}). This trajectory requires ODIN to realize a list of about 60°. Due to compensation for the restoring moments in pitch and roll, we can not maximize the heave control over the trajectory. And, in yaw there is never any significant force to overcome, thus it will never show a maximized control output. Overall, we note that this $(STPP)_2$ strategy resembles the pure motion control strategy shown earlier. In Figure 8 we display the experimental and theoretical evolution of the vehicle. We modify the theoretical evolution by an initial rotation to match the orientation ODIN started with in the pool. Before analyzing this experimental result, we would like emphasize again that we are completely in an open loop framework. All controls are computed before the experiment. First note that the evolution of the yaw does not match the prescribed strategy. This is nearly impossible to fix it since there are minimal restoring forces acting in this direction. Any misappropriated thrust will result in a parasite yaw torque. This significantly affects the yaw evolution since there is nothing to counter act it. Parasite torques exist for roll and pitch but are far less prob-

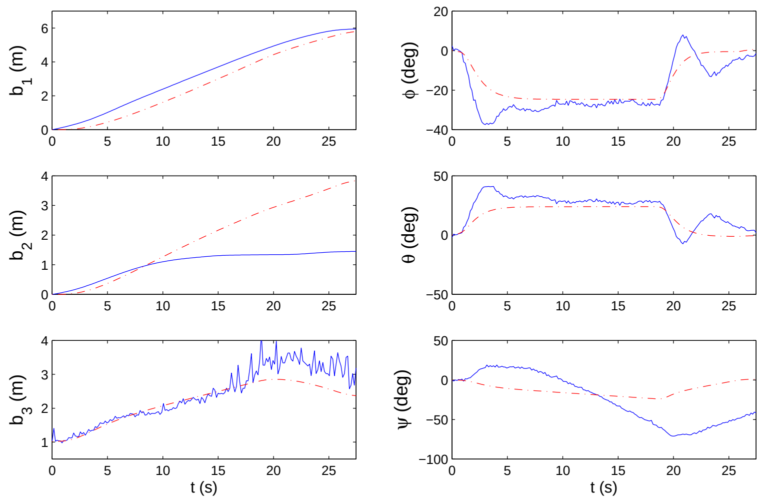

Fig. 8. *Experimental (solid) and theoretical (dashed) evolutions of the AUV for the* $(STPP)_2$ *strategy ending at* $\eta_T = (6, 4, 1, 0, 0, 0)$.

lematic. The restoring moments make any parasite torques nearly negligible. Speaking of the evolution of ϕ and θ, we see that the general trend of the prescribed motion is respected up to the transient behavior. During the transient behavior we can notice rather important *overshooting* that have three main reasons. The first reason is the overshooting of the thrusters. This indicates that our 0.9 s linear junction still does not erase the transient response of the thrusters. However, using a larger duration for the junction would render the $(STPP)_p$ control strategy less time efficient and might even impair its convergence. The second identified reason is the sensor accuracy and transient behavior. As for the yaw, we cannot rule out that the sensor are prone to misreading when confronted to fast change of orientation. A final reason would be an over estimation of the drag coefficient in pitch and roll (and then in yaw by symmetry). Indeed the observed overshooting looks very similar to what would happen to a less damped system. Future experiments will try to determine the respective impact of those three issues and to cope with them. The depth evolution (b_3) looks fine until about 20 s when ODIN dove more than expected. This is caused by the instability of the thrusters around a switching along with some overshooting based on inaccurate drag estimates. We are also examining the effect of the tether on the buoyancy of ODIN. For the horizontal motions, we note a good comparison in surge, while the sway evolution leaves much to be desired. However, with the overshooting issue in pitch and roll, we can not expect to match both surge and sway evolutions with the prescribed motion. For the $(STPP)_2$ trajectory, we have found that the drag of the tether may not be small as first expected.

6 Conclusion

Bridging the gap between theory and application is the ultimate goal of control theory. During the past two decades, major developments have occurred in the field of geometric control. The work presented here is based on recent progress made in geometric optimal control with the objective to develop an efficient tool for scientists such as oceanographers. Our underwater vehicle application successfully demonstrates the applicability of these geometric methods to design trajectories for a concrete applications. In the reverse, the underwater vehicles presents an ideal platform to extend the theory to mechanical systems with dissipative and potential forces. In particular, we are interested in the extension of the notion of kinematic reductions. For the next step, we are currently studying the addition of energy consumption as a minimization cost. Since we can determine the optimal time between two configurations at rest, we can consider the trade-off between energy consumption and duration of the trajectory. Trajectories which minimize energy consumption can be computed using similar numerical techniques as presented for time minimization in this paper. Moreover, implementation will be straightforward since the control strategy design is the same.

Acknowledgment

We would like to take time to thank a few people without whom this research would not succeed. We thank Bruce Kennard, the UH pool coordinator, for his ongoing flexibility and willingness to fit us into an already busy schedule. Chris M^{c}Leod has been beneficial to the project with his knowledge and expertise in electrical engineering and general maintenance and upkeep of ODIN. Also, thank you to Jeff Fines for his devotion to the mundane filming and film analysis of all the experiments which we conduct.

References

[1] Volker Bertram and Alberto Alvarez. *Hydrodynamic Aspects of AUV Design. 5th International Conference on Computer Applications and Information Technology in the Maritime Industries* Leiden/Netherlands, 2006.

[2] B. Bonnard and M. Chyba. *Singular Trajectories and their Role in Control Theory. Springer-Verlag*, 2003.

[3] R.W. Brocket. *Minimum Attention Control*, In *36th IEEE Conf. on Decision and Control*, San Diego, 1997.

[4] R.W. Brocket. *Minimum Attention Control in a Motion Control Context*, In *42th IEEE Conf. on Decision and Control*, Maui, 2003.

[5] F. Bullo and A.D. Lewis. *Geometric Control of Mechanical Systems Modeling, Analysis, and Design for Simple Mechanical Control Systems*. Springer-Verlag, New York-Heidelberg-Berlin, 49 in Texts in Applied Mathematics, 2004.

[6] M.Chyba, T. Haberkorn. R.N. Smith. G.R. Wilkens. *Geometric Properties of Singular Extremals for a Submerged Rigid Body*, Preprint, 2007.

[7] M. Chyba, T. Haberkorn, R.N. Smith, Scott Weatherwax and Song K. Choi. "Experimental Analysis of a Theoretical AUV Model". *Submitted 26th International Conference on Offshore Mechanics and Artic Engineering (OMAE)*, San Diego, 2007.

[8] M. Chyba and T. Haberkorn. "Autonomous Underwater Vehicles: Singular Extremals and Chattering". *Proceedings of the 22nd IFIP TC 7 Conference on System Modeling and Optimization*, Italy, 18-22 July 2005.

[9] M. Chyba and T. Haberkorn. "Designing Efficient Trajectories for Underwater Vehicles Using Geometric Control Theory". *Proceedings of the 24rd International Conference on Offshore Mechanics and Arctic Engineering*, Greece, 12-17 June 2005.

[10] M. Chyba, N.E. Leonard and E.D. Sontag. "Singular Trajectories in the Multi-Input Time-Optimal Problem: Application to Controlled Mechanical Systems". *Journal on Dynamical and Control Systems* 9(1):73-88, 2003.

[11] M. Chyba. "Underwater Vehicles: A Surprising Non Time-Optimal Path".In *42th IEEE Conf. on Decision and Control*, Maui 2003.

[12] M.Chyba, H. Maurer, H.J. Sussmann and G. Vossen. "Underwater Vehicles: The Minimum Time Problem". In *Proceedings of the 43th IEEE Conf. on Decision and Control*, Bahamas, 2004.

[13] T.I. Fossen. "Guidance and Control of Ocean Vehicles". *Wiley*, New York, 1994

[14] R. Fourer, D.M. Gay and B.W. Kernighan. "AMPL: A Modeling Language for Mathematical Programming". *Duxbury Press*, Brooks-Cole Publishing Company,1993.

[15] C.Y. Kaya and J.L. Noakes. "Computation Method for Time-Optimal Switching Control". *Journal of Optimization Theory and Applications*, 117(1):69-92, 2003.

[16] E. Hairer, S.P. Norsett and G. Wanner."Solving Ordinary Differential Equations I. Nonstiff Problems. 2nd edition".*Springer series in computational mathematics*, Springer Verlag, 1993.

[17] N.E. Leonard. "Stability of a Bottom-Heavy Underwater Vehicle". *Automactica*, 33(3):331-46, 1997.

[18] N.E. Leonard. *Stabilization of Steady Motions of an Underwater Vehicle*, Proc. of the 35th IEEE Conference on Decision and Control, 1996.

[19] F.H. Imlay. "The Complete Expressions for Added Mass of a Rigid Body Moving in an Ideal Fluid". Technical Report DTMB, 1961.

[20] R. Bhattacharyya. "Dynamics of Marine Vehicles", Wiley, 1978.

[21] E. Allmendinger. "Submersible Vehicle Design", SNAME, 1990.

[22] Malcolm A. MacIver, Ebraheem Fontaine and Joel W. Burdick. "Designing Future Underwater Vehicles: Principles and Mechanisms of the Weakly Electric Fish". In *IEEE Journal of Oceanic Engineering*, Volume 29, No. 3, July 2004.

[23] H. Maurer, C. Büskens, J-H.R. Kim and I.C. Kaya. "Optimization Methods for Numerical Verification of Second Order Sufficient Conditions for Bang-Bang Controls". *Optimal Control Application and Methods*, 26:129-156, 2005.

[24] Levente Molnar, Edin Omerdic and Daniel Toal. "Hydrodynamic Aspects of AUV Design". *Ocean 2005 Europe IEEE Conference* Brest, France, June 2005.

[25] A. Waechter and L. T. Biegler. "On the Implementation of an Interior-Point Filter-Line Search Algorithm for Large-Scale Nonlinear Programming". *Research Report RC 23149, IBM T.J. Watson Research Center*, Yorktown, New-York.